ЛАБОРАТОРНАЯ РАБОТА №1

«Исследование характеристик локальной сети с помощью утилит ОС Linux»

Лабораторная работа должна выполняться в проводной локальной вычислительной системе лаборатории со звездообразной топологией на базе маршрутизатора (роутера). Убедитесь в наличии в исследуемой операционной системе Linux утилит ifconfig, netstat, nmap (netmap), nc (netcat) и tcpdump. При отсутствии этих и других нужных утилит в вашем дистрибутиве Linux с помощью менеджера пакетов установите их из репозитория. Задания исследования также можно выполнять с использованием утилит дистрибутива Kali Linux.

Внимание! В качестве исключения исследования можно выполнить в произвольной домашней сети на основе роутера. В этом случае в начале отчета по работе должна быть приведена топология исследуемой сети со всеми хостами ( IP-номерами).

Теоретические сведения. Методические рекомендации

**Nmap** — это сканер портов с открытым исходным кодом для Linux® и для многих других платформ, помогающий существенно укрепить безопасность вашей системы. Nmap — это мощный инструмент, который можно использовать в различных целях, например, для сканирования состояния безопасности или для обнаружения серверов в сети. Продукт имеет открытый исходный код и доступен бесплатно для всех основных платформ, включая операционные системы Linux, IBM® AIX®, Mac OS X и Microsoft® Windows®. Дальше описываются основные возможности Nmap и рассматриваются некоторые способы использования этого инструмента.

**Внимание!**

Соблюдайте осмотрительность при экспериментах с инструментом Nmap и при его практическом применении. Нередко считается, что применение Nmap является враждебным действием, поскольку этот инструмент можно использовать для зондирования сети с целью выявления уязвимостей в системе защиты.

Установка

В большинстве дистрибутивов Linux инструмент Nmap доступен в виде пакета в репозитории соответствующего дистрибутива. Например, в среде Debian (и в родственных средах, таких как Ubuntu) для установки Nmap выполните команду apt-get install nmap. В среде Red Hat Enterprise Linux для установки Nmap достаточно выполнить команду yum install nmap..

Указание хостов или сетей для сканирования

В командной строке инструмента Nmap должен быть указан список хостов или сетей, с которыми он должен работать. Для этого имеется несколько опций.

1. Список отдельных имен серверов или IP-адресов. Примеры: "server1" — сканирование только хоста с именем server1; "192.168.0.240" — сканирование только хоста с IP-адресом 192.168.0.240.
2. Список всей сети в формате CIDR. Примеры: "192.168.0.0/24" — сканирование 256 адресов в диапазоне от 192.168.0.0 до 192.168.0.255; "10.0.0.0/8" – сканирование 16 млн. адресов в диапазоне от 10.0.0.0 до 10.255.255.255.
3. Диапазон IP-адресов. Примеры: "192.168.0.50-95" — сканирование 46 IP-адресов в диапазоне от 192.168.0.50 до 192.168.0.95.
4. Использование флага -iL для указания текстового файла, содержащего список хостов или сетей для сканирования.
5. Использование группового символа \*. Примеры: 192.168.0.\* — сканирование 256 адресов в диапазоне от 192.168.0.0 до 192.168.0.255.
6. Расширенные опции, позволяющие просканировать сеть на основе имени хоста. Примеры: "example.com/24" — сканирование подсети класса C любой сети, в которой присутствует хост example.com.

В командной строке можно указать несколько хостов в разных форматах. Примеры: запись "server1 192.168.0.6 10.0.0.0/8" означает хост server1, IP-адрес 192.168.0.6 и 16 млн. адресов в сети 10.0.0.0/8.

Верификация заданных хостов

Соблюдайте максимальную осторожность; сканируйте только те хосты и IP-адреса, на сканирование которых у вас есть разрешение. Чтобы получить список объектов для сканирования, выполните команду nmap -n -sL, сопровождаемую списком хостов или сетей. Флаг -n дает указание не выполнять обратный запрос имен, а опция -sL указывает Nmap вывести список IP-адресов, которые будут просканированы. Таким образом, команда nmap -n -sL ничего не посылает в сеть; она просто выводит список объектов, которые инструмент Nmap просканировал бы в случае реального сканирования.

Рекомендуется выполнить эту команду перед началом сканирования, чтобы убедиться в том, что вы будете сканировать только те хосты, которые запланированы для изучения. В примере на рис. 1 заданы хосты "linux1" и "192.168.0.200-210". Опции -n -sL заставляют инструмент Nmap — не осуществляя контакта ни с одним из хостов — вывести список из 12 адресов хостов, подлежащих сканированию.

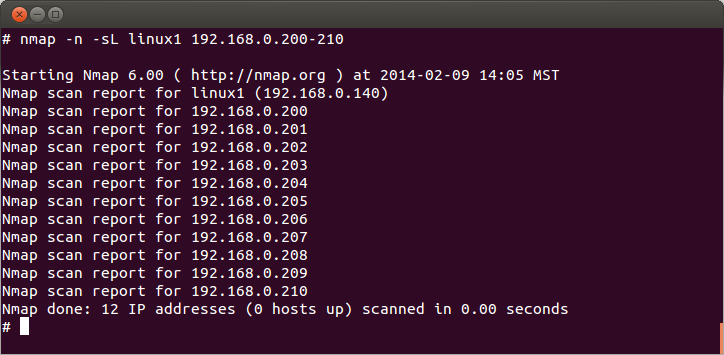


Рисунок 1. Верификация заданных хостов перед выполнением реального сканирования

Обнаружение серверов в сети

С точки зрения администратора одним из самых полезных аспектов инструмента Nmap является его способность обнаруживать серверы или хосты в сети. Эта функциональность полезна для аудита существующих хостов, для документирования ИТ-среды, для сбора информации о подсетях, которые почти исчерпали набор доступных IP-адресов и т. д.

Одна из простейших операций Nmap — обратный запрос имен для диапазона IP-адресов. Например, чтобы увидеть, имеют ли какие-либо хосты в подсети 192.168.0.0/24 записи типа reverse DNS, можно выполнить команду nslookup для каждого из этих 256 адресов — или же осуществить всю эту работу с помощью инструмента Nmap. С этой целью просто задайте флаг-sL, который дает инструменту Nmap указание вывести список IP-адресов и разрешенных имен хостов, подлежащих сканированию. Соответствующий пример приведен на рис. 2.

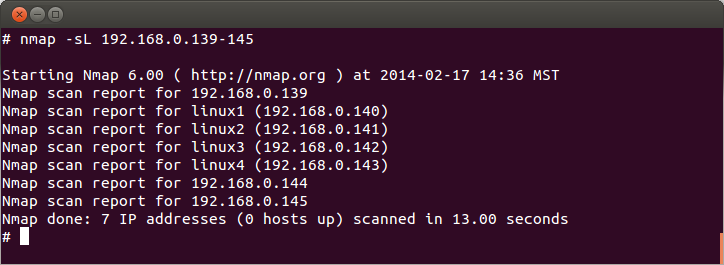


Рисунок 2. Демонстрация DNS-имен серверов

Как показано на рис. 2, если вы укажете инструменту Nmap осуществить обратный запрос DNS-имен для семи адресов в диапазоне от 192.168.0.139 до 192.168.0.145, Nmap сделает это и покажет соответствующие имена в виде своей выходной информации. Несколько IP-адресов не имеют DNS-имен, поэтому в списке они представлены без имен.

*Ping-сканирование*— это удобная функция для определения того, сколько IP-адресов используется и доступно в адресном пространстве, а также для инвентаризации хостов, присутствующих в сети. Ping-сканирование подобно выполнению команды ping для одного IP-адреса с целью проверки его реакции, однако инструмент Nmap способен сделать это в более крупном масштабе и быстро.

При выполнении ping-сканирования Nmap отправляет ICMP-пакет (Internet Control Message Protocol) (ping), а также запрашивает порты 80 и 443 (которые широко используются веб-серверами). Если IP-адрес отвечает на какой-либо из этих запросов, то соответствующий хост считается функционирующим. Для выполнения ping-сканирования с помощью Nmap задайте флаг -sn. На рис. 3 показаны результаты ping-сканирования для адресов 192.168.0.245 — 192.168.0.250. Инструмент Nmap обнаружил, что некоторые из этих хостов функционируют, и сообщил об этом. Обратите внимание, что, если вы осуществляете сканирование в той же подсети, в которой находитесь сами, то Nmap также сообщает MAC-адрес системы и даже говорит, на какую компанию зарегистрирован этот MAC-адрес. Эта функциональность помогает определить, какие аппаратные средства используются в системе.

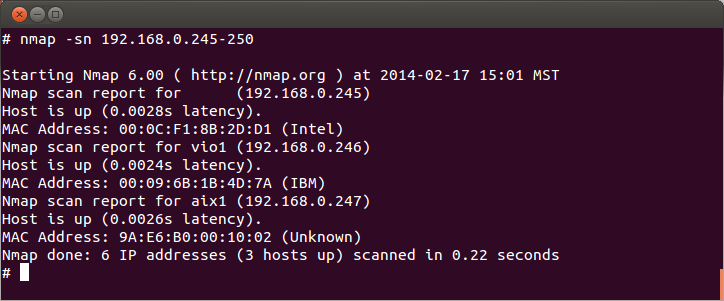


Рисунок 3. Ping-сканирование с помощью Nmap

Базовое сканирование портов

Nmap - это в первую очередь сканер портов. Каждый сетевой сервис в системе "прослушивает" сетевые порты в диапазоне от 1 до 65535. Например, сервис SSH (Secure Shell) прослушивает порт 22, а HTTP-сервис (Hypertext Transfer Protocol) веб-сервера прослушивает порт 80. Сканер портов, такой как Nmap, пытается установить соединения с портами по IP-адресу с целью обнаружения работающих сетевых сервисов.

Сканер портов очень полезен потому, что один из важнейших принципов хорошей защиты системы состоит в отключении неиспользуемых сервисов. Каждый исполняющийся на сервере сервис - это потенциальная цель для атаки. Каждый день выявляются все новые уязвимости системы безопасности, при этом весьма часто причина состоит в уязвимости определенного сетевого сервиса. Если вам не нужен какой-либо сетевой сервис - отключите его! Nmap помогает установить, какие сервисы работают на одиночном хосте или на каждом хосте вашей сети.

Например, если в вашей сети для доступа к серверам используется сервис SSH, вам следует отключить сервис Telnet. В результате пользователи не смогут обращаться к сервису Telnet, который отправляет пароли в виде открытого текста. Кроме того, если в Telnet будет выявлена новая уязвимость, она не повлияет на вашу сеть, если этот сервис не исполняется.

Самый простой способ осуществления сканирования портов с помощью Nmap - выполнить команду nmap, сопровождаемую списком имен серверов, IP-адресов или диапазонов IP-адресов (см. рис. 4).

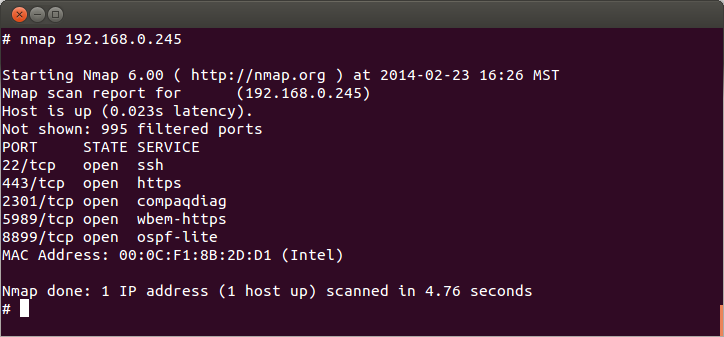


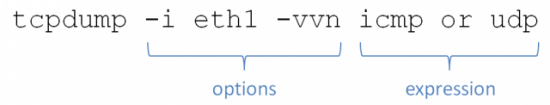
Рисунок 4. Базовое сканирование портов с помощью Nmap

На рис. 4 показаны результаты базового сканирования портов с помощью Nmap для IP-адреса 192.168.0.245. В случае выполнения этой команды без каких-либо опций, Nmap сканирует 1000 портов, которые используются чаще других. Вы можете дать инструменту Nmap указание проверить все 65535 портов на хосте с данным IP-адресом; однако это занимает очень много времени, поэтому по умолчанию Nmap сканирует лишь 1000 наиболее используемых портов. Как показано на рис. 4, инструмент Nmap нашел пять открытых портов по данному IP-адресу (22, 442, 2301, 5989, 8899).

Инструмент Nmap предоставляет десятки опций, позволяющих контролировать практически все аспекты выполнения сканирования портов. Например, опция -p0- задает сканирование всех 65535 портов вместо сканирования 1000 портов по умолчанию.

Кроме того, имеются опции для управления типом сканирования, скоростью сканирования и т. д. Дополнительную информацию о других опциях можно получить в man-руководстве по Nmap.

**Tcpdump** - удобный сетевой анализатор. Естественно, что для получения максимальной информации при работе с tcpdump, просто необходимо иметь представления о стеке протоколов TCP/IP.

Понимание протокола TCP/IP дает широкое пространство для использования анализатора и устранения неисправностей и неполадок в работе сети за счет разбора пакетов. Ниже приводятся несколько опций, которые помогут наиболее полно и подробно использовать эту утилиту.

Команда tcpdump состоит из двух частей: опций и выражения для фильтра

Выражение для фильтра определяет, какие пакеты захватывать, а опции - какие из них показывать в выводе, также опции отвечают за поведение утилиты.

Первый используемый ключ –n, который запрещает попытки преобразования адресов в доменные имена, тем самым выдавая нам чистые IP адреса с портами.

Второй это –X, который задает для каждого пакета вывод как hex- (шестнадцатеричная система), так и ASCII-содержимого пакета.  
И последняя опция это –S, которая заменяет вывод относительной TCP нумерации, на абсолютную. Смысл этой замены состоит в том, что при относительной нумерации некоторые проблемы могут скрыться от вашего внимания.

Нужно понимать, что основное преимущество tcpdump перед другими утилитами заключается в том, что в ней возможен подробный и ручной разбор пакетов. Также нужно помнить, что по умолчанию tcpdump использует только первые 68 байт пакета, т.е. если необходимо видеть больше, то следует использовать ключ -s number , где number количество байт которые следует захватить. В случае задания number 0 , произойдет полный захват пакета, поэтому лучше использовать значение 1514, что даст полный захват стандартного, для сетей Ethernet, пакета.

Список наиболее часто используемых ключей:

-c : задается проверка размера файла захвата перед каждой очередной записью захваченного пакета, если размер больше, то файл сохраняется и запись идет в новый файл;

-e : выводится ethernet заголовок (канальный уровень) в каждой строке дампа;

-i any : прослушивание всех интерфейсов, на случай если вам необходим весь трафик;

-n : запрещает преобразование адресов в доменные или символьные имена;

-nn : запрещает преобразование адресов и портов в доменные или символьные имена;

-q : краткий вывод информации, за счет уменьшения вывода информации о протоколе;

-X : выводит как hex- так и ASCII-содержимое пакета;

-v, -vv, -vvv : задает вывод дополнительной информации о захваченных пакетах, что дает возможность более широкого анализа.  
 Несколько примеров для использования:

# Стандартный вывод пакетов tcpdump -nS  
# Расширенный стандартный вывод tcpdump -nnvvS  
# Глубокий разбор пакета tcpdump -nnvvXS  
# Наиболее подробная информация о трафике tcpdump -nnvvXSs 1514

Выражения позволяют производить целевое сканирование и задавать типы трафика. Существует три основных типа выражений: type, dir, and proto.  
Опции выражения type бывают host, net и port. Для выражения направления задаваемого dir существующие опции src, dst, src or dst, и src and dst.  
 Несколько стандартных выражений:

host // анализ трафика на основе IP адреса ( также работает с символьными именами, если не задано -n)  
tcpdump host 1.2.3.4src, dst // анализ трафика только для определенного назначения или передатчика  
tcpdump src 2.3.4.5  
tcpdump dst 3.4.5.6 net // захват трафика принадлежащего определенной сети  
tcpdump net 1.2.3.0/24 port // анализ трафика с определенного порта (входящего или исходящего)  
tcpdump port 3389 выражения для фильтра  
 Выражения для фильтра есть булевы критерии (да/нет) для оценки совпадения пакета с образцом. Все пакеты, не соответствующие заданным условиям, игнорируются.

Синтаксис пакетного фильтра - мощный и прозрачный. Вначале он содержит ключевые слова, называемые "примитивы", которые представляют собой различные классификаторы для пакетов. Это могут быть: протокол, адрес, порт или направление. Они могут быть соединены между собой в цепочки с помощью операторов и/или, сгруппированы и объединены правилами наследования, отсортированы с помощью отрицания.

Полностью синтаксис описан в man-странице pcap-filter, здесь же представлены некоторые примеры:

|  |
| --- |
| tcp  port 25 and not host 10.0.0.3  icmp or arp or udp  vlan 3 and ether src host aa:bb:cc:dd:ee:ff  arp or udp port 53  icmp and \(dst host mrorange or dst host mrbrown\) |

Подобно опциям выражения для фильтра не являются обязательными. Пустой фильтр означает "все пакеты".

Вывод tcpdump и его расшифровка

Степень понимания вывода tcpdump зависит от того, насколько хорошо вы понимаете особенности работы различных протоколов в рамках модели OSI. Вывод обычно привязан к совпадению пакета с определенным протоколом. Например, если вызвать tcpdump с опциями -t и -n, ARP пакеты будут отображаться примерно так:

|  |
| --- |
| arp who-has 10.0.0.1 tell 10.0.0.2  arp reply 10.0.0.1 is-at 00:01:02:03:04:05 |

ARP - простой протокол, он используется для разрешения IP-адресов в MAC-адреса сетевых карт.

Пакеты DNS, с другой стороны, будут описаны несколько по-другому:

|  |
| --- |
| IP 10.0.0.2.50435 > 10.0.0.1.53: 19+ A? linuxjournal.com. (34)  IP 10.0.0.1.53 > 10.0.0.2.50435: 19 1/0/0 A 76.74.252.198 (50) |

Поначалу это может показаться недостаточно ясным, но по мере изучения того, как работают протоколы различных уровней, наполнится смыслом. DNS намного более сложный протокол, чем ARP, но кроме этого он работает на более высоком уровне. Это означает, что пакеты нижележащих протоколов также отображаются в выводе. В отличие от ARP, который не маршрутизируется между различными физическими сегментами сети, DNS это протокол для всего Интернета. Для маршрутизации этих пакетов используется уровень IP, для транспорта задействован UDP. Это делает DNS протоколом пятого уровня (IP - третий уровень, UDP - четвертый).

Информация уровней UDP/IP, содержащая адрес и порт источника, отображается в левой стороне строки, специфическая DNS-информация - в правой. Несмотря на то, что синтаксис довольно сжатый, он достаточен для определения базовых элементов DNS.

Просмотрите секцию "OUTPUT FORMAT" справки man для tcpdump для полного описания всех зависимых от протоколов форматов вывода. Пакеты некоторых протоколов видны в выводе лучше, другие хуже, но важная информация обычно находится легко.

Захват вывода в файл

Помимо обычного режима с выводом информации на консоль (стандартный вывод), tcpdump также поддерживает режим записи вывода в файл. Режим активируется опцией -w, в которой задается путь к файлу.

При записи в файл tcpdump использует другой формат, чем при выводе на экран. Это так называемый сырой вывод, в нем не производится первичный анализ пакета. Эти файлы можно затем использовать в сторонних программах типа Wireshark, потому что формат записей в файл соответствует универсальному формату "pcap" (на ввод tcpdump такой файл можно подать с помощью опции -r). Эта возможность позволяет нам захватывать пакеты на одной машине, а анализировать на другой.

Анализ протоколов на основе TCP

Тcpdump хорошо работает с протоколами, основанными на работе с отдельным пакетом, например, IP, UDP, DHCP, DNS и ICMP. Если же есть некий "поток", или последовательность пакетов для установления соединения, tcpdump не сможет напрямую анализировать эти потоки и сценарии соединений. Такие протоколы, как HTTP, SMTP и IMAP с точки зрения сетевого взаимодействия гораздо больше похожи на интерактивные приложения, чем "пакетные" протоколы.

TCP прозрачно для пользователя обрабатывает все низкоуровневые детали, необходимые для сеансов связи в рамках сессионных протоколов. Здесь происходит инкапсуляция данных, ориентированных на поток, в пакеты (сегменты), которые уже могут быть посланы по сети. Все такие подробности скрыты ниже уровня приложений. Поэтому для захвата пакетов протоколов, ориентированных на соединения, требуются дополнительные шаги. Поскольку каждый сегмент TCP есть кусочек данных уровня приложения, информация о нем не может быть использована напрямую. Чтобы это имело бы смысл, нужно полностью восстанавливать TCP-сессию (поток) из последовательности отдельных пакетов. Этой возможности tcpdump не имеет.

Практическое задание

1. Внимательно прочитайте задание и справки по синтаксису основных команд Linux по справочнику команд, который рассматривался в 7 семестре.

2. Зарегистрируйтесь в системе в первой консоли с правами администратора.

3. Зарегистрируйтесь во второй консоли с правами пользователя.

4. Из консоли администратора с помощью команды ifconfig -a выведите на экран данные о текущем состоянии всех сетевых интерфейсов компьютера. Какую информацию из прочитанного вывода вы извлекли?

Запомните, как обозначается основной Ethernet-адаптер (он может обозначаться eth0, eth1, eth2), и в дальнейшем используйте в сетевых командах это имя. Далее в тексте задания он упоминается как eth0.

5. Выведите информацию о сетевых интерфейсах с помощью команды netstat –ai, сравните возможности двух использованных утилит.

6. Командой ps –ef | more выведите список процессов и убедитесь, что сетевой процесс inetd работает. Иначе его нужно запустить вручную командой inetd. Если исследуемая версия ОС не содержит сервера telnet (по причине его явной уязвимости некоторые версии Linux не предусматривают использования этого протокола), соответствующие пункты задания выполните с защищенной программной оболочкой Secure Shell (SSH).

7. Отключите сетевой адаптер командой ifconfig eth0 down (см. справку по сетевым командам). Присвойте сетевому интерфейсу временный MAC-адрес A0:B1:C2:D3:E4:N, где N – двузначный номер вашего компьютера.

Подключите адаптер к сети. Убедитесь в том, что его аппаратный адрес изменен.

8. Назначьте основному сетевому интерфейсу компьютера временный IP-адрес и маску подсети. Для этого введите команду ifconfig eth0 192.168.0.N netmask 255.255.255.0, где N – номер компьютера. Повторным вводом команды ifconfig eth0 убедитесь в том, что запись введенной информации произведена.

Присвоенные сетевые адреса будут действовать до перезагрузки компьютера.

9. Проверьте работоспособность петли обратной связи, послав на свой же компьютер эхо-запрос ping 127.0.0.1. Убедившись, что отклики поступают, остановите зондирование.

10. Присвойте сетевому адаптеру дополнительный IP-адрес 192.168.0.20+N, где N – номер компьютера. Проверьте прохождение ICMP-пакетов между сетевыми адресами на локальном компьютере.

11. Организуйте сеанс telnet на собственном компьютере, используя для этого интерфейс обратной петли или дополнительный IP-адрес (если необходимо, установите из сетевого репозитория пакет telnet в своей операционной системе).

Для этого перейдите в консоль пользователя, наберите команду telnet localhost и после сообщения об успешном соединении введите login и пароль администратора. Почему вам было отказано в доступе?

12. Еще раз установите сеанс telnet через петлю обратной связи, используя на этот раз учетную запись обычного пользователя. После установления сеанса просмотрите список каталогов и файлов в нескольких директориях, список процессов и убедитесь, что в «удаленном» режиме доступа вы можете выполнять все команды, которые доступны пользователю, зарегистрированному на удаленном узле.

13. Перейдите в консоль администратора и с помощью команды w или who посмотрите, сколько сейчас пользователей в системе, кто они и с каких терминалов работают. Обратите внимание на то, как обозначаются локальный и удаленный терминалы.

14. С помощью команды netstat –a проконтролируйте список запущенных сервисов и их состояние.

15. Вернитесь в консоль пользователя и завершите локальный сеанс telnet командой exit. Получите сообщение о закрытии сетевого соединения. Проверьте эту информацию с помощью команды netstat.

16. Попробуйте войти на один из компьютеров сети в сеансе telnet (учетные записи пользователей на всех компьютерах должны быть одинаковы). Что потенциально опасного вы можете сделать на удаленном компьютере?

17. Войдите на произвольный узел по протоколу Secure Shell (команда ssh с указанием IP-адреса хоста, после запроса необходимо ввести пароль администратора). Проверьте свои возможности по манипуляции удаленным компьютером.

18. С помощью команды arp –a посмотрите таблицу соответствия сетевых и аппаратных адресов. Где расположен ARP-кэш и почему он сейчас пуст?

19. С помощью утилиты ping исследуйте локальную сеть, к которой подключен ваш компьютер в диапазоне адресов, идентифицирующих конкретный компьютер в сети, от 1 до 15 (192.168.0.1/15). Объясните, в чем неудобство такого метода сканирования.

20. Проверьте, обновилась ли после сканирования динамическая ARP-таблица. Если она содержит нужную вам информацию о сети, ее можно сохранить в файле командой arp –a > /home/arp1 (через несколько минут информация о сетевых узлах будет изменена, и если тот или иной сетевой узел не проявляет активности, данные о нем в кэше будут утрачены). С помощью команды arp –s <IP-адрес> <MAC-адрес> создайте статическую arp-таблицу. Выясните местоположение этой таблицы.

21. С помощью утилиты nmap исследуйте локальную сеть, к которой подключен ваш компьютер. Что вы можете сказать о полученной информации?

22. Отключите ответы своего сетевого адаптера на ARP-запросы других хостов командой ifconfig eth0 –arp. С помощью команды ifconfig eth0 убедитесь, что настройка выполнена.

23. Выждите несколько минут для сброса ARP-таблиц на компьютерах локальной сети и с одного из компьютеров сети с помощью утилиты ping попробуйте обнаружить отклик своего компьютера. Достаточно ли надежно защищает компьютер от сканирования данная мера? Насколько нарушается при этом возможность работы в сети? Включите arp-отклик командой ifconfig eth0 arp.

24. Отключите сетевой интерфейс на своем компьютере с помощью команды ifconfig eth0 down. Повторите попытку обнаружения своего компьютера с одного из соседних узлов. Можете ли вы сами при этом проявлять какую-либо сетевую активность? Сделайте выводы. Вновь включите сетевой адаптер с помощью команды ifconfig eth0 up.

25. Переведите сетевой адаптер своего компьютера в режим перехвата всех пакетов с помощью команды ifconfig eth0 promisc. С помощью команды ifconfig eth0 убедитесь, что настройка выполнена. При этом сетевой адаптер превращается в устройство подслушивания, но одновременно он становится уязвимым к сетевым атакам на отказ в обслуживании.

26. Ознакомьтесь с синтаксисом команды tcpdump.

27. С помощью утилиты перехватите и прочитайте сетевые пакеты:

- отправленные из одного определенного адреса,

- являющиеся результатом сетевого обмена между двумя хостами сети.

28. Запишите несколько перехваченных пакетов в файл и просмотрите их в шестнадцатеричном коде (приведите для примера скриншот с 4-5 перехваченными пакетами).

Найдите характерные поля и идентификаторы в заголовках канального, сетевого и транспортного уровней. Определите аппаратные и сетевые адреса, номера портов, иные характерные признаки, идентифицирующие сетевые протоколы.

Представьте преподавателю отчет о проделанной работе с выводами по пунктам.

Контрольные вопросы

1. Как закрепить за одним сетевым адаптером несколько IP–адресов?

2. Как программным путем изменить аппаратный адрес сетевой карты?

3. Можно ли перехватывать трафик без установленного IP–адреса?

4. Для чего нужно отключать ARP–отклик?

5. Где находится ARP–кэш? Как долго хранятся в нем данные?

6. Перечислите известные вам виды сетевого сканирования.